

Amendments to the Claims

1. (Currently Amended) In a device including plural applications requesting remote resources, a method of managing overlapping connection sessions created to support access to remote resources by said plural applications, the method comprising: receiving, in a connection manager that is separate from the plural applications, a connection request comprising a process identification and saving the process identification in a data structure;

establishing a connection, using the connection manager, upon receiving a connection request when no connection exists;

receiving a disconnection request comprising a the process identification and removing the disconnecting process identification from the data structure; and

terminating the connection upon receiving a disconnection request when no process identifications remain in the data structure after removing the disconnecting process identification.

2. (Currently Amended) The ~~device~~ method of claim 1 further comprising: saving a time that a connection request was received;

after a threshold period after the time, removing a process identification from the data structure if a process associated with the process identification has terminated; and

terminating the connection when no process identifications remain in the data structure after removing the identification of the terminated process.

3. (Currently Amended) The ~~device~~ method of claim 1 wherein the device is a computer and the connection is a dial-up connection.

4. (Currently Amended) The ~~device~~ method of claim 1 wherein the device is a wireless device and the connection is a wireless connection.

5. (Currently Amended) The ~~device~~ method of claim 4 wherein the wireless device is a telephone.

6. (Currently Amended) The ~~device~~ method of claim 5 wherein the wireless device is a hand held computer.
7. (Currently Amended) A computerized method comprising:
 - receiving, from a first application, a first request, in a connection manager, for a connection to a remote resource;
 - saving in a data structure, an identifier of the first request for a connection;
 - upon receiving a the first request for connection, creating the connection between the first application and the remote resource when the connection is not already established;
 - receiving, in the connection manager, a second request from a second application for connection to the same remote resource as the first application;
 - sharing the connection to the remote resource between the first application and the second application;
 - receiving a request for a disconnection from a remote resource;
 - deleting from the data structure, an identifier of the request for the disconnection;
 - disconnecting the connection upon a disconnection request when the deleted identifier is the last identifier of a request for a connection in the data structure.
8. (Original) The method of claim 7 further comprising:
 - removing an identifier of a request for a connection from the data structure after a period of time after the request is made if a process associated with the identifier has terminated.
9. (Original) The method of claim 7 wherein a request for a connection originates from an application and the remote resource is a web server.
10. (Original) The method of claim 9 wherein the connection is a dial-up connection between a modem and an Internet service provider.

11. (Original) The method of claim 7 wherein the method is running on a wireless device with plural applications sending the connection requests and communicating with remote resources over the connection.
12. (Original) A computer system comprising:
 - a processor coupled to memory and a hardware device for communicating with remote resources;
 - software in memory and comprising:
 - an operating service for receiving system service requests via an application services interface;
 - plural applications requesting remote services from the operating service via the application services interface;
 - a connection manager for establishing via the hardware device a connection shared by plural applications communicating with remote resources over the connection and for maintaining the connection when an application requests a disconnection while another application is still using the connection.
13. (Original) The system of claim 12 wherein the connection manager disconnects the connection when a last application using the connection calls disconnect.
14. (Original) The system of claim 12 wherein the connection manager maintains a list of applications that have requested the connection.
15. (Original) The system of claim 14 wherein the connection manager disconnects the connection when an application requests a disconnect and no other application is on the list.
16. (Original) The computer system of claim 12 wherein the system is a personal computer.
17. (Original) The computer system of claim 12 wherein the connection is wireless.

18. (Currently Amended) A computer-readable medium comprising executable instructions for performing a method comprising:

creating a connection when a process request communicating with remote resources requiring the connection;

using a connection manager, storing identifiers of processes requesting communicating with remote resources via the connection;

using the connection manager, removing an identifier of a process from the stored identifiers when the process requests a disconnection;

maintaining the connection when a process requests a disconnection when stored identifiers indicate another process is communicating with remote resources via the connection; and

disconnecting the connection when a process requests a disconnection when stored identifiers indicate no other process is communicating with remote resources via the connection.

19. (Original) The computer-readable medium of claim 18 further comprising executable instructions for removing an identifier of a process from the stored identifiers when the process has terminated.

20. (Original) The computer-readable medium of claim 18 further comprising executable instructions for periodically removing identifiers of processes from the stored identifiers when the processes have terminated without requesting a disconnect.